



## ServerChecker

- port: std::string
- keyDB: std::map<std::string, std::string>
- +ServerChecker(port: std::string)
- +~ServerChecker()
- +start()
- initWSA(wsaData: WSADATA&)
- initServerSocket(): SOCKET
- startListening(listener: SOCKET&)
- acceptNewClient(listener: SOCKET&): SOCKET
- read(client: SOCKET&, input: std::string&)
- write(client: SOCKET&, toSend: const std::string&)
- close(client: SOCKET&)
- validate (data: std::string&)
- fillDB(): std::map<std::string, std::string>
- ok(): std::string
- deny(): std::string
- bad(): std::string



## MyServerChecker

- +MyServerChecker(port: std::string)
- fillDB(): std::map<std::string, std::string>
- ok(): std::string
- deny(): std::string
- bad(): std::string

## ClientChecker

- port: std::string
- host: std::string
- +ClientChecker(host: std::string, port: std::string)
- +~ClientChecker()
- +execute(key: const std::string&, user: const std::string&)
- initWSA(wsaData: WSADATA&)
- initClientSocket(): SOCKET
- write(client: SOCKET&, toSend: const std::string&)
- read(client: SOCKET&, toFill: std::string&)
- close(client: SOCKET&)
- handle(data: const std::string&)



## MyClientChecker

- +MyClientChecker(host: std::string, port: std::string)
- handle(data: const std::string&)